THE UNIVERSITY
OF WISCONSIN
madison, wisconsin

COMPUTER ANALYSIS OF THREE-FACTOR INTERACTIONS IN CONTINGENCY TABLES

Marvin A. Kastenbaum and Dennis Kuba

MRC Technical Summary Report #636
April 1966

UNITED STATES ARMY

MATHEMATICS RESEARCH CENTER

MATHEMATICS RESEARCH CENTER, UNITED STATES ARMY
THE UNIVERSITY OF WISCONSIN

Contract No.: DA-11-022-ORD-2059

COMPUTER ANALYSIS OF THREE-FACTOR INTERACTIONS IN CONTINGENCY TABLES

Marvin A. Kastenbaum and Dennis Kuba

MRC Technical Summary Report #636
April 1966

Madison, Wisconsin

ABSTRACT

Numerous methods have been proposed for testing the hypothesis of zero three-factor interaction in contingency tables. These methods are incorporated into a single computer program which is designed to calculate the test statistic by any or all of the proposed techniques, and which also provides estimators of the interactions and corresponding two-sided confidence intervals. The program is written in FORTRAN 63 for the CDC-3600 computer, and will analyze interactions in contingency tables of dimensions $2 \leq r \leq 5$, $2 \leq s \leq 5$, $2 \leq t \leq 16$, for $r \leq s \leq t$.

COMPUTER ANALYSIS OF THREE-FACTOR INTERACTIONS IN CONTINGENCY TABLES

Marvin A. Kastenbaum and Dennis Kuba

1. Introduction

The analysis of three-factor interactions in contingency tables has been the subject of numerous recent papers culminating with the admirably lucid treatment and summary given by Goodman [15]. This series of papers begins with one by Kastenbaum and Lamphiear [8] in which the authors present an iterative technique for solving $(r-1)(s-1)(t-1)$ simultaneous fourth-degree equations. These equations result from an extension Bartlett's [2] test of zero interaction in a $2 \times 2 \times 2$ table to the general three-way $(r \times s \times t)$ table as presented by Roy and Kastenbaum [6]. Subsequent authors [10, 11, 12, 13, 14, 15] have proposed alternative techniques of analysis, each progressively simpler than the preceding one, but all testing the same hypothesis. In every case; the test statistic is distributed asymptotically as chi-square with $(r-1)(s-1)(t-1)$ degrees of freedom.

It is the purpose of this paper to describe a computer program which has been designed to calculate the test statistic by any or all of the proposed techniques. This program also provides the estimates of interaction and corresponding simultaneous confidence intervals given by Goodman [15]. The program was written in FORTRAN 63 for the CDC-3600 computer, and will do analyses on contingency tables of dimensions $2 \leq r \leq 5$, $2 \leq s \leq 5$, $2 \leq t \leq 16$, for $r \leq s \leq t$. [Appendix].

2. Tests of Hypotheses

Let n_{ijk} be the number of observations in the ith row, jth column, and

Sponsored by the Mathematics Research Center, United States Army, Madison, Wisconsin, under Contract No.: DA-11-022-ORD-2059.

kth layer of a three-way contingency such that $\sum_{i=1}^{r}\sum_{j=1}^{s}\sum_{k=1}^{t} n_{ijk} = n$, the total sample size. Denote by $0 < \Pi_{ijk} < 1$ the corresponding probability where $\sum_{i=1}^{r}\sum_{j=1}^{s}\sum_{k=1}^{t} \Pi_{ijk} = 1$. Also define the following relationships:

$$\sum_{i=1}^{r} n_{ijk} = n_{.jk},\ \sum_{j=1}^{s} n_{ijk} = n_{i.k},\ \sum_{k=1}^{t} n_{ijk} = n_{ij.} \quad ,$$

$$\sum_{i=1}^{r}\sum_{j=1}^{s} n_{ijk} = n_{..k},\ \sum_{i=1}^{r}\sum_{k=1}^{t} n_{ijk} = n_{.j.},\ \sum_{j=1}^{s}\sum_{k=1}^{t} n_{ijk} = n_{i..} \quad ,$$

with similar relationships for the Π_{ijk}'s . The hypothesis of zero three-factor interaction in an $r \times s \times t$ contingency table is given by:

$$H_0: \frac{\Pi_{rsk}\Pi_{ijk}}{\Pi_{isk}\Pi_{rjk}} = \frac{\Pi_{rst}\Pi_{ijt}}{\Pi_{ist}\Pi_{rjt}} \tag{2.1}$$

for all integers i, j, k such that $1 \leq i < r,\ 1 \leq j < s,\ 1 \leq k < t$.

2.1. Kastenbaum and Lamphiear [8]

For the $r \times s \times t$ table, Kastenbaum and Lamphiear generalized an iterative technique proposed by Norton [4] for handling the simultaneous fourth-degree equations which arise in the estimation process. Under the null hypothesis (2.1), estimates of the parameters may be achieved by first solving for all x_{ijk} in the following systems of equations:

$$\frac{(n_{rsk} - \sum_{i=1}^{r-1}\sum_{j=1}^{s-1} x_{ijk})(n_{ijk} - x_{ijk})}{(n_{isk} + \sum_{j=1}^{s-1} x_{ijk})(n_{rjk} + \sum_{i=1}^{r-1} x_{ijk})} = \frac{(n_{rst} - \sum_{i=1}^{r-1}\sum_{j=1}^{s-1} x_{ijt})(n_{ijt} - x_{ijt})}{(n_{ist} + \sum_{j=1}^{s-1} x_{ijt})(n_{rjt} + \sum_{i=1}^{r-1} x_{ijt})} , \tag{2.1.1}$$

for all positive integers $i < r$, $j < s$, and $k < t$, and subject to the constraints

$\sum_{k=1}^{t-1} x_{ijk} = -x_{ijt}$ for all positive integers $i \leq r$ and $j \leq s$. Let ν_{ijk} be the values of x_{ijk} which satisfy (2.1.1) for all integers $1 \leq i \leq r$, $i \leq j \leq s$, and $1 \leq k \leq t$. Then

$$X^2 = \sum_{i=1}^{r} \sum_{j=1}^{s} \sum_{k=1}^{t} \nu_{ijk}^2/(n_{ijk} - \nu_{ijk}) \qquad (2.1.2)$$

is distributed asymptotically as chi-square with $(r-1)(s-1)(t-1)$ degrees of freedom.

The program solves equations (2.1.1) by applying Newton's method of functional iteration to a technique proposed by Norton [4], and computes the value of the test statistic (2.1.2). It then prints out the number of iterations, the number of degrees of freedom, the value of the test statistic, and the observed and expected cell frequencies, with identification for all the cells in the contingency table. A sample print-out is given in Table 1.

2.2. Darroch [10]

An alternative solution to equations (2.1.1) is given by Darroch. This method involves the iterative solution for δ_{jk}, φ_{ki}, and ψ_{ij} in the following $(rs + rt + st)$ simultaneous non-linear equations:

$$\frac{n_{.jk}}{n} = \delta_{jk} \sum_{i=1}^{r} \varphi_{ki} \psi_{ij} \qquad (2.2.1)$$

$$\frac{n_{i.k}}{n} = \varphi_{ki} \sum_{j=1}^{s} \psi_{ij} \delta_{jk} \qquad (2.2.2)$$

$$\frac{n_{ij.}}{n} = \psi_{ij} \sum_{k=1}^{t} \delta_{jk} \varphi_{ki} \quad . \qquad (2.2.3)$$

The iteration begins by setting $\varphi_{ki} = \varphi_{ki}^{(1)} = n_{i.k}/n_{..k}$ and $\psi_{ij} = \psi_{ij}^{(1)} = n_{ij.}/n_{i..}$ in equation (2.2.1) and solving for $\delta_{jk} = \delta_{jk}^{(2)}$, then in equation (2.2.2) let

$\psi_{ij} = \psi_{ij}^{(1)}$ and $\delta_{jk} = \delta_{jk}^{(2)}$ and solve for $\phi_{ki} = \varphi_{ki}^{(2)}$. In equation (2.2.3), let $\delta_{jk} = \delta_{jk}^{(2)}$ and $\varphi_{ki} = \varphi_{ki}^{(2)}$ and solve for $\psi_{ij} = \psi_{ij}^{(2)}$. Then return to equation (2.2.1), letting $\varphi_{ki} = \varphi_{ki}^{(2)}$ and $\psi_{ij} = \psi_{ij}^{(2)}$, and solve for $\delta_{jk} = \delta_{jk}^{(3)}$. Continue the iteration in this way until $\psi_{ij}^{(m)} = \psi_{ij}^{(m-1)}$, $\delta_{jk}^{(m)} = \delta_{jk}^{(m-1)}$, and $\varphi_{ki}^{(m)} = \varphi_{ki}^{(m-1)}$ to five decimal places for all positive integers $i \leq r$, $j \leq s$, $k \leq t$. At this point calculate

$$X^2 = \sum_{i=1}^{r} \sum_{j=1}^{s} \sum_{k=1}^{t} [n_{ijk} - n\delta_{jk}^{(m)} \varphi_{ki}^{(m)} \psi_{ij}^{(m)}]^2 / n\delta_{jk}^{(m)} \varphi_{ki}^{(m)} \psi_{ij}^{(m)} . \qquad (2.2.4)$$

Darroch has shown that this test statistic is distributed asymptotically as chi-square with $(r-1)(s-1)(t-1)$ degrees of freedom. Moreover (2.2.4) and (2.1.2) are algebraically identical.

The program solves equations (2.2.1), (2.2.2), and (2.2.3), and computes the value of the test statistic (2.2.4). The print-out is identical to that of Table 1, except that the number of iterations is generally smaller. In this sense, the Darroch procedure is superior to the Kastenbaum-Lamphiear procedure.

2.3. Plackett [11]

Let $\theta_{ijk} = \Pi_{ijk}/\Pi_{..k}$ be the conditional probability that an observation will fall in the ith row and jth column, given that it is in the kth layer. It follows that the null hypothesis, (2.1), may be rewritten as

$$H_0: \Delta_{ijk} = \Delta_{ijt} \qquad (2.3.1)$$

for all integers $1 \leq i < r$, $1 \leq k < t$, where $\Delta_{ijk} = \theta_{rsk}\theta_{ijk}/\theta_{isk}\theta_{rjk}$. Using a criterion suggested by Woolf [5], Plackett presents a procedure for testing a null hypothesis equivalent to (2.3.1), namely

$$H_0: \Gamma_{ijk} = \Gamma_{ijt} \qquad (2.3.2)$$

for all integers $1 \leq i < r$, $1 \leq j < s$, $1 \leq k < t$, where $\Gamma_{ijk} = \log \Delta_{ijk}$. This procedure is based on the fact that the maximum likelihood estimator of $\log \Delta_{ijk}$ is $\log d_{ijk}$, where $d_{ijk} = n_{ijk} n_{rsk} / n_{isk} n_{rjk}$ for all integers $1 \leq i < r$, $1 \leq j < s$, $1 \leq k < t$. Moreover the variance of $\log d_{ijk}$ may be estimated consistently by

$$u_{ijk} = \frac{1}{n_{ijk}} + \frac{1}{n_{rsk}} + \frac{1}{n_{isk}} + \frac{1}{n_{rjk}} \quad .$$

If R and S are two matrices of order $(r-1) \times r$ and $(s-1) \times s$ respectively with rows orthogonal to each other and to the unit vector, then the direct product, $[R*S]$, is a matrix with $(r-1)(s-1)$ rows and rs columns. The elements of each row of this matrix provide the coefficients of a linear combination of the logarithm of the frequencies in the kth layer of the contingency table. More specifically, the matrices R and S are formed as follows:

$$R = \{\rho_{\alpha i}\} = \begin{cases} 1 & \text{for } 1 \leq i \leq \alpha , \\ -\alpha & \text{for } i = \alpha + 1 , \\ 0 & \text{for } i > \alpha + 1 , \end{cases} \quad \text{where } 1 \leq i \leq r,\ 1 \leq \alpha < r, \text{ and}$$

$$S = \{\sigma_{\beta j}\} = \begin{cases} 1 & \text{for } 1 \leq j \leq \beta , \\ -\beta & \text{for } j = \beta + 1 , \\ 0 & j > \beta + 1, \end{cases} \quad \text{where } 1 \leq j \leq s,\ 1 \leq \beta < s \ .$$

Then for each positive integer $k \leq t$, a column vector z_k, is generated from the product of the $(r-1)(s-1) \times rs$ matrix $[R*S]$ with the $rs \times 1$ vector $\{\log n_{ijk}\}$.

The elements of z_k are

$$z_{k\alpha\beta} = \sum_{i=1}^{r} \sum_{j=1}^{s} \rho_{\alpha i} \, \sigma_{\beta j} \log n_{ijk} \quad . \qquad (2.3.3.)$$

The asymptotic distribution of $z_k = \{z_{k\alpha\beta}\}$ is multivariate normal, with dispersion matrix

$$V_k = [R*S] \, D_{ijk}^{-1} \, [R*S]' \ ,$$

where $D^{-1}_{n_{ijk}}$ is a square matrix of order rs with elements $\{\frac{1}{n_{ijk}}\}$ on the diagonal and zeros elsewhere. The elements of V_k are

$$\mathrm{Cov}[z_{k\alpha\beta}, z_{k\alpha'\beta'}] = \sum_{i=1}^{r} \sum_{j=1}^{s} \rho_{\alpha i}\, \rho_{\alpha' i}\, \sigma_{\beta j}\, \sigma_{\beta' j}/n_{ijk} \,, \qquad (2.3.4)$$

for all positive integers $\alpha, \alpha' < r$ and $\beta, \beta' < s$. If z'_k denotes the transpose of z_k, and V_k^{-1} the inverse of V_k, then on the hypothesis of zero three-factor interaction, (2.3.2), the statistic

$$Y^2 = \sum_{k=1}^{t} z'_k V_k^{-1} z_k - [\sum_{k=1}^{t} z'_k V_k^{-1}][\sum_{k=1}^{t} V_k^{-1}]^{-1}[\sum_{k=1}^{t} V_k^{-1} z_k] \qquad (2.3.5)$$

is distributed asymptotically as chi-square with $(r-1)(s-1)(t-1)$ degrees of freedom.

This portion of the program computes the elements of all the vectors z_k and of their associated dispersion matrices V_k from formulas (2.3.3) and (2.3.4) for all integers $1 \leq k \leq t$. To evaluate the test statistic, (2.3.5), $(t+1)$ square matrices of order $(r-1)(s-1)$ are inverted. The print-out, Table 2, displays all the vectors z_k and their associated dispersion matrices V_k for all integers $1 \leq k \leq t$; and $\sum_{k=1}^{t} L_k^2 = \sum_{k=1}^{t} z'_k V_k^{-1} z_k$, $P = [\sum_{k=1}^{t} V_k^{-1}]^{-1}$, $\sum_{k=1}^{t} z'_k V_k^{-1}$, as well as the value of the test statistic, (2.3.5), and the number of degrees of freedom.

2.4. Goodman [14, 15]

The approach suggested by Goodman modifies Plackett's method for testing the null hypothesis (2.3.2) by redefining the two matrices R and S as follows:

$$R = \{\rho_{\alpha i}\} = \begin{cases} 1 & \text{for } i = \alpha \,, \\ -1 & \text{for } i = r \,, \\ 0 & \text{elsewhere, where } 1 \leq i \leq r,\ 1 \leq \alpha < r, \text{ and} \end{cases}$$

$$S = \{\sigma_{\beta j}\} = \begin{cases} 1 & \text{for } j = \beta \\ -1 & \text{for } j = s \\ 0 & \text{elsewhere, where } 1 \leq j \leq s,\ 1 \leq \beta < s \ . \end{cases}$$

As in Plackett's method, column vectors $g_k = \{g_{k\alpha\beta}\}$ are generated, where

$$g_{k\alpha\beta} = \sum_{i=1}^{r} \sum_{j=1}^{s} \rho_{\alpha i}\, \sigma_{\beta j} \log n_{ijk} \ . \qquad (2.4.1)$$

The asymptotic distribution of each g_k is multivariate normal with dispersion matrix U_k, whose elements are

$$\mathrm{Cov}[g_{k\alpha\beta},\, g_{k\alpha'\beta'}] = \sum_{i=1}^{r} \sum_{j=1}^{s} \rho_{\alpha i}\, \rho_{\alpha' i}\, \sigma_{\beta j}\, \sigma_{\beta' j} / n_{ijk} \qquad (2.4.2)$$

for all positive integers $\alpha, \alpha' < s$. If $U_k^{-1} = M_k$, $Q = [\sum_{k=1}^{t} M_k]^{-1}$, and $\tilde{g} = \sum_{k=1}^{t} M_k\, g_k$, then the statistic

$$Y^2 = \sum_{k=1}^{t} g_k'\, M_k\, g_k - \tilde{g}'\, Q \tilde{g} \qquad (2.4.3)$$

is distributed asymptotically as chi-square with $(r-1)(s-1)(t-1)$ degrees of freedom. Moreover, expressions (2.3.5) and (2.4.3) are identical.

Except for the new definitions of the matrices R and S, the computational procedure using either Plackett's or Goodman's methods appear to be identical. In both methods it is necessary to invert $(t + 1)$ matrices of order $(r-1)(s-1)$. However, Goodman makes a significant contribution, at this point, by demonstrating that only one $(r-1)(s-1)$ matrix and t matrices of order $(r-1)$, for $r \leq s$, need to be inverted in calculating the test statistic (2.4.3). The reduction in computing time which results from this modified procedure may be appreciable. The calculations are carried out as follows:

(i) For every integer k, $1 \leq k \leq t$, define the $(r-1) \times (r-1)$ matrices

$$B^{(k)} = D_{n_{isk}} - \frac{1}{n_{.sk}} \{n_{isk}\}\{n_{isk}\}' ,$$

$$B_j^{(k)} = D_{n_{ijk}} - \frac{1}{n_{.jk}} \{n_{ijk}\}\{n_{ijk}\}' ,$$

where $D_{n_{isk}}$ and $D_{n_{ijk}}$ are $(r-1) \times (r-1)$ diagonal matrices with elements n_{isk} and n_{ijk} respectively, and where $\{n_{isk}\}'$ and $\{n_{ijk}\}'$ are the respective transposes of the $(r-1) \times 1$ column vectors $\{n_{isk}\}$ and $\{n_{ijk}\}$, for integers $1 \leq i < r$, and $1 \leq j < s$.

(ii) Evaluate $C^{(k)} = B^{(k)} + \sum_{j=1}^{s-1} B_j^{(k)}$, an $(r-1) \times (r-1)$ matrix, and its inverse $G^{(k)}$, $1 \leq k \leq t$.

(iii) Construct the submatrices

$$M_{jj'}^{(k)} = \begin{cases} B_j^{(k)} - B_j^{(k)} G^{(k)} B_j^{(k)} & \text{for } j = j' = 1, 2, \ldots, s-1 , \\ - B_j^{(k)} G^{(k)} B_{j'}^{(k)} & \text{for } j \neq j' . \end{cases}$$

(iv) For every k, $1 \leq k \leq t$, form the square matrix M_k of order $(r-1)(s-1)$ in equation (2.4.3) from the $(s-1)^2$ submatrices $M_{jj'}^{(k)}$ each of order $(r-1) \times (r-1)$.

(v) Evaluate the test statistic using equation (2.4.3) .

The print-out (Table 3), for this section of the program displays all the vectors g_k, their associated dispersion matrices U_k, and $\sum_{k=1}^{t} H_k^2 = \sum_{k=1}^{t} g_k' M_k g_k$, Q, $\mathfrak{g}'$, as well as the value of the test statistic, (2.4.3), and the number of degrees of freedom.

2.5. Goodman [15]: The $2 \times 2 \times t$ Contingency Table

For the three-way contingency table with $r = 2$ rows, $s = 2$ columns, and $t \geq 2$ layers, Goodman proposes three alternative test statistics. Two of these are based on an analysis of the cell frequencies, and the third, which is a special case of the procedure discussed in section 2.4, is based on an analysis of the log-frequencies.

(i) Define $d_k = \dfrac{n_{11k} n_{22k}}{n_{12k} n_{21k}}$, $u_k = \dfrac{1}{n_{11k}} + \dfrac{1}{n_{22k}} + \dfrac{1}{n_{12k}} + \dfrac{1}{n_{21k}}$, $v_k = d_k^2 u_k$,

and $w_k = 1/v_k$, $1 \leq k \leq t$. Then

$$X^2 = \sum_{k=1}^{t} d_k^2 w_k - [\sum_{k=1}^{t} d_k w_k]^2 / \sum_{k=1}^{t} w_k \tag{2.5.1}$$

is distributed asymptotically as chi-square with $(t-1)$ degrees of freedom. Equation (2.5.1) may be used to test the hypothesis of zero three-factor interaction which, in the $2 \times 2 \times t$ table may be specified as

$$H_0: \frac{\Pi_{111} \Pi_{221}}{\Pi_{121} \Pi_{211}} = \frac{\Pi_{11k} \Pi_{22k}}{\Pi_{12k} \Pi_{21k}} \qquad 2 \leq k \leq t \ . \tag{2.5.2}$$

Equation (2.5.2) may be rewritten as

$$H_0: \Delta_1 = \Delta_k, \quad 2 \leq k \leq t, \tag{2.5.3}$$

where
$$\Delta_k = \frac{\Pi_{11k} \Pi_{22k}}{\Pi_{12k} \Pi_{21k}} = \frac{\theta_{11k} \theta_{22k}}{\theta_{12k} \theta_{21k}}$$

is a measure of the two factor interaction in the kth layer of the table. The maximum likelihood estimator of Δ_k is d_k, and its variance can be estimated consistently by v_k.

Moreover, Goodman points out that this null hypothesis may be partitioned into the following $(t-1)$ sub-hypotheses:

$$H_1:\ \Delta_1 = \Delta_2$$

$$H_2:\ \Delta_1 = \Delta_3\ \ (\text{given that } \Delta_1 = \Delta_2)$$

$$H_3:\ \Delta_1 = \Delta_4\ \ (\text{given that } \Delta_1 = \Delta_2 = \Delta_3) \tag{2.5.4}$$

$$\vdots$$

$$H_{t-1}:\ \Delta_1 = \Delta_t\ \ (\text{given that } \Delta_1 = \Delta_2 = \ldots = \Delta_{t-1}) \ .$$

Each of these sub-hypotheses H_k $(1 \leq k < t)$ can be tested using a single degree of freedom as follows:

$$\text{Define } d_\gamma = \frac{n_{11\gamma} n_{22\gamma}}{n_{12\gamma} n_{21\gamma}} \qquad 1 \leq \gamma < k \ ,$$

and corresponding quantities u_γ, v_γ, and w_γ. Let $d_k^* = \sum_{\gamma=1}^{k} d_\gamma w_\gamma / \sum_{\gamma=1}^{k} w_\gamma$ and $v_k^* = 1/\sum_{\gamma=1}^{k} w_\gamma$. Then to test H_k, $1 \leq k < t$, in equation (2.5.4) compute

$$X_k^2 = [d_{k+1} - d_k^*]^2 / [v_{k+1} + v_k^*] \ . \tag{2.5.5}$$

The sum, $\sum_{k=1}^{t-1} X_k^2$ is equal to X^2 given by (2.5.1).

When H_k is true, (2.5.5) is distributed asymptotically as chi-square with one degree of freedom. Also when H_0 is true, each of the $(k-1)$ statistics X_k^2 will have asymptotically independent chi-square distributions each with one degree of freedom.

(ii) Let $b_k = \dfrac{1}{d_k}$, $a_k = b_k^2 u_k$, $h_k = \dfrac{1}{a_k}$, $1 \leq k \leq t$,

$$b_\gamma = \frac{1}{d_\gamma},\quad a_\gamma = b_\gamma^2 u_\gamma,\quad h_\gamma = \frac{1}{a_\gamma},\quad 1 \leq \gamma \leq k \ ,$$

$$b_k^* = \sum_{\gamma=1}^{k} b_\gamma h_\gamma / \sum_{\gamma=1}^{k} h_\gamma, \text{ and } a_k^* = 1/\sum_{\gamma=1}^{k} h_\gamma \ .$$

Here b_k is the maximum likelihood estimate of $1/\Delta_k$, and its variance can be estimated consistently by a_k. When the null hypothesis (2.5.3) is true

$$Z^2 = \sum_{k=1}^{t} b_k^2 h_k - [\sum_{k=1}^{t} b_k h_k]^2 / \sum_{k=1}^{t} h_k \tag{2.5.6}$$

is distributed asymptotically as chi-square with $(t-1)$ degrees of freedom. Tests of H_k $(1 \leq k < t)$ in equation (2.5.4) are given by

$$Z_k^2 = [b_{k+1} - b_k^*]^2 / [a_{k+1} + a_k^*] \tag{2.5.7}$$

which, when H_k is true, is distributed asymptotically as chi-square with one degree of freedom. Also $\sum_{k=1}^{t-1} Z_k^2 = Z^2$.

(iii) Let

$$g_k = \log d_k, \quad m_k = 1/u_k, \quad 1 \leq k \leq t,$$

$$g_\gamma = \log d_\gamma, \quad m_\gamma = 1/u_\gamma, \quad 1 \leq \gamma \leq k,$$

$$g_k^* = \sum^{k} g_\gamma m_\gamma / \sum_{\gamma=1}^{k} m_\gamma, \quad \text{and} \quad u_k^* = 1/\sum_{\gamma=1}^{k} m_\gamma .$$

Here g_k is the maximum likelihood estimate of $\Gamma_k = \log \Delta_k$, and its variance can be estimated consistently by u_k. The null hypothesis, (2.5.3), becomes

$$H_0: \Gamma_1 = \Gamma_k \quad \text{for} \quad 2 \leq k \leq t . \tag{2.5.8}$$

When (2.5.8) is true

$$Y^2 = \sum_{k=1}^{t} g_k^2 m_k - [\sum_{k=1}^{t} g_k m_k]^2 / \sum_{k=1}^{t} m_k \tag{2.5.9}$$

is distributed asymptotically as chi-square with $(t-1)$ degrees of freedom. This statistic, (2.5.9), is the special form which equation (2.4.3) takes when $r = 2$ and $s = 2$. Tests of hypotheses analogous to (2.5.4) involving the log-frequencies are given by

$$Y_k^2 = [g_{k+1} - g_k^*]^2/[u_{k+1} + u_k^*] \quad , \tag{2.5.10}$$

for every integer k, $1 \leq k < t$, which, when H_k is true, is distributed asymptotically as chi-square with one degree of freedom. Also, $\sum_{k=1} Y_k^2 = Y^2$. Moreover, when H_k is true, Y_k^2 and Z_k^2 are asymptotically equivalent to X_k^2 .

This portion of the program computes all the test statistics given by equations (2.5.1), (2.5.5), (2.5.6), (2.5.7), (2.5.9) and (2.5.10). The print out, displayed in Table 4, gives values of d_k, v_k, b_k, a_k, g_k, and u_k for $1 \leq k \leq t$, and X_k^2, Z_k^2, and Y_k^2 for $1 \leq k < t$. In addition, the program computes and displays the values X^2, Z^2, Y^2, $\sum_{k=1}^{t-1} X_k^2$, $\sum_{k=1}^{t-1} Z_k^2$, $\sum_{k=1}^{t-1} Y_k^2$, and the number of degrees of freedom.

2.6. Kullback, Kupperman, and Ku [12, 13]

The procedures outlined by Plackett [11] and Goodman [14, 15] for testing the hypothesis of zero three-factor interaction, (2.1), are primarily generalizations of a method proposed by Woolf [5] in which the test criterion is based on a logit transformation of the data. This test criterion is one alternative to the Pearson [1] chi-square test of goodness of fit. Another alternative is the likelihood-ratio test criterion proposed by Wilks [3], and investigated by Woolf [7] and Kullback, Kupperman, and Ku [12,13].

Kullback, Kupperman, and Ku resort to an information theory approach and define a minimum discrimination information statistic [M.D.I.S.]. This statistic is distributed asymptotically as chi-square under the null hypothesis, and as noncentral chi-square under the alternative hypothesis, with appropriate degrees

of freedom and noncentrality parameter. It has additive properties in the sense that it can be analyzed into several additive components for a hypothesis which is equivalent to the combination of several hypotheses of interest. Each such component of the M.D.I.S. is itself an M.D.I.S., and is distributed asymptotically as chi-square with appropriate degrees of freedom. Moreover, the M.D.I.S. has the convexity property which is useful in finding other M.D.I.S. under certain restrictions and groupings. In particular, for nonnegative real numbers a_i and b_i, this property yields

$$\sum_{i=1}^{n} a_i \ln(a_i/b_i) \geq \left(\sum_{i=1}^{n} a_i\right) \ln\left(\sum_{i=1}^{n} a_i / \sum_{i=1}^{n} b_i\right) . \qquad (2.6.1)$$

Equality holds if and only if $\frac{a_1}{b_1} = \frac{a_2}{b_2} = \dots = \frac{a_n}{b_n}$.

In the $r \times s \times t$ table, the criterion for testing H_0, (2.1), is given by

$$2\hat{I} = 2 \sum_{i=1}^{r} \sum_{j=1}^{s} \sum_{k=1}^{t} n_{ijk} \log \left[\frac{n_{ijk} n_{i..} n_{.j.} n_{..k}}{n n_{ij.} n_{i.k} n_{.jk}} \right] . \qquad (2.6.2)$$

This statistic is distributed asymptotically as chi-square with $(r-1)(s-1)(t-1)$ degrees of freedom. In any specific example the convexity property may not hold. This may result in a negative value for (2.6.2), in which case Kullback, Kupperman and Ku [12, pp. 225-226] recommend one of the following alternative analyses of the data:

row-layer interaction with column

$$2\hat{I} = 2\sum_{i=1}^{r}\sum_{j=1}^{s}\sum_{k=1}^{t} n_{ijk} \log\left[\frac{n_{ijk}\,n_{.j.}\,y_{i.k}}{n_{ij.}\,n_{.jk}\,n_{i.k}}\right], \qquad (2.6.3)$$

$$\text{where } y_{i.k} = \sum_{j=1}^{s} n_{ij.}\,n_{.jk}/n_{.j.} \;;$$

column-layer interaction with row

$$2\hat{I} = 2\sum_{i=1}^{r}\sum_{j=1}^{s}\sum_{k=1}^{t} n_{ijk} \log\left[\frac{n_{ijk}\,n_{i..}\,y_{.jk}}{n_{ij.}\,n_{.jk}\,n_{i.k}}\right] \qquad (2.6.4)$$

$$\text{where } y_{.jk} = \sum_{i=1}^{r} n_{ij.}\,n_{i.k}/n_{i..} \;;$$

row-column interaction with layer

$$2\hat{I} = 2\sum_{i=1}^{r}\sum_{j=1}^{s}\sum_{k=1}^{t} n_{ijk} \log\left[\frac{n_{ijk}\,n_{..k}\,y_{ij.}}{n_{ij.}\,n_{.jk}\,n_{i.k}}\right] \qquad (2.6.5)$$

$$\text{where } y_{ij.} = \sum_{k=1}^{t} n_{i.k}\,n_{.jk}/n_{..k} \;.$$

The program computes the M.D.I.S. test statistics using equations (2.6.2), (2.6.3), (2.6.4), and (2.6.5), and prints out the corresponding values of $2\hat{I}$ as well as the number of degrees of freedom.

3. Confidence Intervals

Of all the papers on the subject of three-factor interaction in contingency tables, only Goodman's [15] deals with the problem of estimation. Indeed, the final section of this paper is devoted exclusively to a discussion of methods for estimating the magnitude of the three-factor interaction, and for obtaining confidence limits for it.

Let

$$A_{ibjckd} = \log\left\{\frac{\Pi_{ijk}\,\Pi_{bck}}{\Pi_{bjk}\,\Pi_{ick}} \Big/ \frac{\Pi_{ijd}\,\Pi_{bcd}}{\Pi_{bjd}\,\Pi_{icd}}\right\} \tag{3.1}$$

be the measure of a particular three-factor interaction for all integers $1 \le i < b \le r$, $1 \le j < c \le s$, and $1 \le k < d \le t$. Depending on the values of b, c, and d, there can be as many as $\binom{r}{2}\binom{s}{2}\binom{t}{2}$ such three-factor interactions in an $r \times s \times t$ contingency table. In particular, for $b = r$, $c = s$, $d = t$, there are $(r-1)(s-1)(t-1)$ interactions of the form

$$A_{irjskt} = \Gamma_{ijk} - \Gamma_{ijt} = \log(\Delta_{ijk}/\Delta_{ijt})$$

$$= \log\left\{\frac{\Pi_{ijk}\,\Pi_{rsk}}{\Pi_{rjk}\,\Pi_{isk}} \Big/ \frac{\Pi_{ijt}\,\Pi_{rst}}{\Pi_{rjt}\,\Pi_{ist}}\right\} . \tag{3.2}$$

The maximum likelihood estimator of A_{ibjckd} is

$$a_{ibjckd} = \log\left\{\frac{n_{ijk}\,n_{bck}}{n_{bjk}\,n_{ick}} \Big/ \frac{n_{ijd}\,n_{bcd}}{n_{bjd}\,n_{icd}}\right\} , \tag{3.3}$$

and its variance is estimated consistently by

$$S^2_{ibjckd} = u_{ibjck} + u_{ibjcd} , \tag{3.4}$$

where $$u_{ibjck} = \frac{1}{n_{ijk}} + \frac{1}{n_{bck}} + \frac{1}{n_{ick}} + \frac{1}{n_{bjk}} . \tag{3.5}$$

Using these definitions, Goodman proposes three alternative sets of approximate two-sided confidence intervals for A_{ibjckd}:

$$a_{ibjckd} \pm \chi_T(P)\, S_{ibjckd} , \tag{3.6}$$

where $\chi_T^2(P)$ is the $[P \times 100]$th percentile of the chi-square distribution with $T = (r-1)(s-1)(t-1)$ degrees of freedom;

$$a_{ibjckd} \pm \chi_1(P)\, S_{ibjckd} \quad , \tag{3.7}$$

where $\chi_1(P)$ is the $[\frac{1+P}{2} \times 100]$th percentile of the standardized normal distribution; and for a specified subset of interactions, $A_{(w)}$,

$$a_{(w)} \pm \Phi_W(P)\, S_{(w)} \quad , \tag{3.8}$$

where, for any (b, c, d), $w = (b-1)(c-1)(d-1)$ is the number of elements in the subset $(w) = (ibjckd)$; and

$$W = W(B, C, D) = \sum_{b \in B} \sum_{c \in C} \sum_{d \in D} (b-1)(c-1)(d-1) \quad ,$$

where $B, C,$ and D are subsets of the sets of integers I_r, I_s, I_t such that

$$B \subseteq I_r = \{2, 3, \ldots, r\} \quad ,$$

$$C \subseteq I_s = \{2, 3, \ldots, s\} \quad ,$$

$$D \subseteq I_t = \{2, 3, \ldots, t\} \quad ;$$

and $\Phi_W(P)$ is the $(\frac{2W-1+P}{2W} \times 100)$ percentile of the standardized normal distribution. It follows that $1 \leq w \leq W \leq \binom{r}{2}\binom{s}{2}\binom{t}{2}$. In particular, for the subset $(w) = (irjskt)$, $w = W = T = (r-1)(s-1)(t-1)$, and the intervals (3.8) become

$$a_{irjskt} \pm \Phi_T(P)\, S_{irjskt} \quad , \tag{3.9}$$

where $\Phi_T(P)$ is the $(\frac{2T-1+P}{2T} \times 100)$th percentile of the standardized normal distribution.

In the final paragraph of his paper, Goodman [15] discusses the relative merits of the three alternative sets of confidence intervals (3.6), (3.7), and (3.8). His remarks may be summarized as follows:

(i) If any of the intervals (3.6) do not include zero, the null hypothesis, (2.3.2), will be rejected at a significance level $(1-P)$, when tested using (2.4.3).

(ii) When a specific set of W three-factor interactions is of interest, $W \leq \binom{r}{2}\binom{s}{2}\binom{t}{2}$, $T \geq 1$, and when the usual values of P, $[P = .95 \text{ or } .99]$, are used, (3.8) will yield smaller confidence intervals than (3.6).

(iii) The probability is (approximately) at least P that all the $\binom{r}{2}\binom{s}{2}\binom{t}{2}$ intervals (3.6) include the corresponding true values. That is to say, the probability is (approximately) at least P that all the $\binom{r}{2}\binom{s}{2}\binom{t}{2}$ confidence statements associated with (3.6) are correct.

(iv) The length of the confidence interval will be reduced if (3.7) is used in place of (3.6). However, the probability that all the confidence statements are correct will also be reduced. The expected proportion of correct confidence statements of type (3.7) will be approximately P.

(v) To insure that all confidence statements are correct, with probability (approximately) at least P, use form (3.6). If it suffices to insure that the expected proportion of correct confidence statements is P, then (3.7) should replace (3.6).

(vi) The same consequences will result when intervals (3.7) are used in place of intervals (3.8).

For a prescribed set of integers (b, c, d), and for all (i, j, k) in the range $1 \leq i < b \leq r$, $1 \leq j < c \leq s$, $1 \leq k < d \leq t$, this portion of the program calculates the estimates of interaction and their estimated variances using equations (3.3) and (3.4). Moreover, it evaluates the three alternative sets of approximate two-sided confidence intervals given by equations (3.6), (3.7), and (3.8), for values of $P = 0.95$ and 0.99. Values of $\chi_T^2(P)$ in (3.6) are calculated using the Fisher-Cornish [9] approximation.

Finally, the program is designed to transform the estimates of interaction and the corresponding confidence intervals to their original scale by taking the anti-logarithm of each of the computed values. The results of these calculations are displayed in logarithmic units in Table 5.

[redacted]

REFERENCES

1. Pearson, Karl [1900], "On the criterion that a given system of deviation from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling," Philos. Mag. Series 5, vol. 50, pp. 157-172.

2. Bartlett, M. S., [1935], "Contingency table interactions," Jour. Roy. Soc. Supplement, vol. 2, pp. 248-252.

3. Wilks, S. S. [1935], "The likelihood test of independence in contingency tables," Ann. Math. Stat., vol. 6, pp. 190-196.

4. Norton, H. W., [1945], "Calculation of chi-square for complex contingency tables," Jour. Amer. Stat. Assoc., vol. 40, pp. 251-258.

5. Woolf, Barnet [1955], "On estimating the relation between blood group and disease," Ann. Hum. Gen., vol. 19, pp. 251-253.

6. Roy, S. N. and M. A. Kastenbaum, [1956], "On the hypothesis of no 'interaction' in a multiway contingency table," Ann. Math. Stat., vol. 27, pp. 749-757.

7. Woolf, Barnet, [1957], "The log likelihood ratio test (The G-Test). Methods and tables for tests of heterogeneity in contingency tables," Ann. Hum. Gen., vol. 21, pp. 397-409.

8. Kastenbaum, M. A. and D. E. Lamphiear [1959], "Calculation of chi-square to test the no three-factor interaction hypothesis," Biometrics, vol. 15, pp. 107-115.

9. Fisher, R. A. and E. A. Cornish [1960], "The percentile points of the distributions having known cumulants," Technometrics, vol. 2, pp. 209-225.

10. Darroch, J. N. [1962], "Interactions in multi-factor contingency tables," Jour. Roy. Stat. Soc., Series B, vol. 24, pp. 251-263.

11. Plackett, R. L. [1962], "A note on interactions in contingency tables," Jour. Roy. Stat. Soc., Series B, vol. 24, pp. 162-166.

12. Kullback, S., M. Kupperman, and H. H. Ku [1962], "An application of information theory to the analysis of contingency tables with a table of $2N \ln N$, $N = 1(1)10,000$," Jour. Res. Nat'l. Bur. Stds. B, vol. 66, pp. 217-243.

13. Kullback, S., M. Kupperman, and H. H. Ku [1962], "Tests for contingency tables and Markov chains," Technometrics, vol. 4, pp. 573-608.

14. Goodman, Leo A. [1963], "On Plackett's test for contingency table interactions," Jour. Roy. Stat. Soc., Series B, vol. 25, pp. 179-188.

15. Goodman, Leo A. [1964], "Simple methods for analyzing three-factor interaction in contingency tables," Jour. Amer. Stat. Assoc., vol. 59, pp. 319-352.

Appendix

Program HYPOINT

```
      PROGRAM HYPOINT
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      COMMON/FLAGS/OPTION(10),LOGFLAG
      TYPE INTEGER OPTION
      TYPE INTEGER OUTPUT
      INPUT=60
      OUTPUT=61
   10 CALL READIN
      IF(OPTION(1).EQ.0) GO TO 20
      CALL KASTENBM
   20 IF(OPTION(2)+OPTION(3).EQ.0) GO TO 30
      CALL DARROCH
   30 IF(OPTION(4).EQ.0) GO TO 40
      CALL PLACKETT
   40 IF(OPTION(5).EQ.0) GO TO 50
      CALL GOODMAN1
   50 IF(OPTION(6).EQ.0) GO TO 60
      CALL GOODMAN2
   60 IF(OPTION(7).EQ.0) GO TO 70
      CALL TWOBYTWO
   70 IF(OPTION(8).EQ.0) GO TO 80
      CALL KKK
   80 IF(OPTION(9).EQ.0) GO TO 90
      LOGFLAG=1
      CALL INTRVALS
   90 IF(OPTION(10).EQ.0) GO TO 100
      LOGFLAG=0
      CALL INTRVALS
  100 GO TO 10
      END
```

```
      SUBROUTINE KASTENBM
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      DIMENSION EXP(400),BIJ(16),CIJ(16)
      TYPE INTEGER OUTPUT
 2000 FORMAT(1H1,10A8)
 2001 FORMAT(32H0KASTENBAUM-LAMPHIEAR PROCEDURE )
 2002 FORMAT(////,12H ITERATION   ,8X,18HDEGREES OF FREEDOM,8X,
     *   14H    CHI-SQUARE,15X,2H R,8X,2H S,8X,2H T)
 2003 FORMAT(//////,8X,13H     CELL    ,8X,18HOBSERVED FREQUENCY,8X,
     *   18HEXPECTED FREQUENCY /,1H )
 2004 FORMAT(/3X,I4,18X,I5,15X,F15.6,7X,3(8X,I2))
 2005 FORMAT(6X,3(2X,I3),8X,F15.6,11X,F15.6)
 2006 FORMAT(25H0ITERATION COUNT EXCEEDS I3)
      WRITE OUTPUT TAPE OUTPUT,2000,IDENT
      WRITE OUTPUT TAPE OUTPUT,2001
      DELX=1.0E-5
      ITERSTOP=100
      NRST=IR*JS*KT
      DO 10 N=1,NRST
      EXP(N)=OBS(N)
   10 CONTINUE
      IR1=IR-1
      JS1=JS-1
      NX=IR1*JS1*KT
      ITERS=0
   20 NXZERO=0
      ITERS=ITERS+1
      DO 60 I=1,IR1
      DO 50 J=1,JS1
      SUMC=0.0
      SUMBC=0.0
      DO 30 K=1,KT
      KPART=IR*(JS*K-JS-1)
      IJSK=KPART+IR*JS+I
      EXPISK=EXP(IJSK)
      IRJK=KPART+IR*J+IR
      EXPRJK=EXP(IRJK)
      IRJSK=KPART+IR*JS+IR
      EXPRSK=EXP(IRJSK)
      IJK=KPART+IR*J+I
      EXPIJK=EXP(IJK)
      CIJK=1.0/EXPISK+1.0/EXPRJK+1.0/EXPRSK+1.0/EXPIJK
      CIJK=1.0/CIJK
      BIJK=EXPISK*EXPRJK/EXPRSK/EXPIJK
```

```
      SUMC=SUMC+CIJK
      SUMBC=SUMBC+CIJK*BIJK
      CIJ(K)=CIJK
      BIJ(K)=BIJK
   30 CONTINUE
      HIJ=SUMC/SUMBC
      DO 40 K=1,KT
      KPART=IR*(JS*K-JS-1)
      XIJK=CIJ(K)*(1.0-HIJ*BIJ(K))
      IF(XIJK.LE.DELX) NXZERO=NXZERO+1
      IRJSK=KPART+IR*JS+IR
      EXP(IRJSK)=EXP(IRJSK)-XIJK
      IJK=KPART+IR*J+I
      EXP(IJK)=EXP(IJK)-XIJK
      IJSK=KPART+IR*JS+I
      EXP(IJSK)=EXP(IJSK)+XIJK
      IRJK=KPART+IR*J+IR
      EXP(IRJK)=EXP(IRJK)+XIJK
   40 CONTINUE
   50 CONTINUE
   60 CONTINUE
      IF(ITERS.GE.ITERSTOP) GO TO 200
      IF(NXZERO.LT.NX) GO TO 20
      CHISQ=0.0
      DO 70 N=1,NRST
      EXPN=EXP(N)
      CHISQ=CHISQ+(OBS(N)-EXPN)*(OBS(N)-EXPN)/EXPN
   70 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2003
      IJK=0
      DO 80 K=1,KT
      DO 80 J=1,JS
      DO 80 I=1,IR
      IJK=IJK+1
      WRITE OUTPUT TAPE OUTPUT,2005,I,J,K,OBS(IJK),EXP(IJK)
   80 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2002
      WRITE OUTPUT TAPE OUTPUT,2004,ITERS,NDEGREES,CHISQ,IR,JS,KT
      GO TO 300
  200 WRITE OUTPUT TAPE OUTPUT,2006,ITERSTOP
  300 END
```

```
      SUBROUTINE DARROCH
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      COMMON/SUMS/OBSR(80),OBSS(80),OBST(25),OBSRS(16),OBSTR(5),
     *   OBSST(5),OBSRST
      COMMON/FLAGS/OPTION(10),LOGFLAG
      DIMENSION EXP(400),THETA(80),PHI(80),PSI(25)
      TYPE INTEGER OPTION
      TYPE INTEGER OUTPUT
 2000 FORMAT(1H1,10A8)
 2001 FORMAT(19H0DARROCH PROCEDURE )
 2002 FORMAT(////,12H ITERATION   ,8X,18HDEGREES OF FREEDOM,8X,
     *   14H     CHI-SQUARE,15X,2H R,8X,2H S,8X,2H T)
 2003 FORMAT(//////,8X,13H      CELL     ,8X,18HOBSERVED FREQUENCY,8X,
     *   18HEXPECTED FREQUENCY /,1H )
 2004 FORMAT(/3X,I4,18X,I5,15X,F15.6,7X,3(8X,I2))
 2005 FORMAT(6X,3(2X,I3),8X,F15.6,11X,F15.6)
 2006 FORMAT(25H0ITERATION COUNT EXCEEDS I3)
      WRITE OUTPUT TAPE OUTPUT,2000,IDENT
      WRITE OUTPUT TAPE OUTPUT,2001
      ITERSTOP=100
      TOLERSQ=0.00005*0.00005
      CALL GETSUMS
C
      DO 140 I=1,IR
      OBSSTI=OBSST(I)
      DO 120 J=1,JS
      IJ=I+IR*(J-1)
      PSI(IJ)=OBST(IJ)/OBSSTI
  120 CONTINUE
      DO 130 K=1,KT
      KI=K+KT*(I-1)
      PHI(KI)=OBSS(KI)/OBSRS(K)
  130 CONTINUE
  140 CONTINUE
C
      ITERS=0
  200 ISTOP=1
      ITERS=ITERS+1
C
      DO 230 K=1,KT
      DO 220 J=1,JS
      JK=J+JS*(K-1)
      SUMI=0.0
      DO 210 I=1,IR
      KI=K+KT*(I-1)
      IJ=I+IR*(J-1)
      SUMI=SUMI+PHI(KI)*PSI(IJ)
  210 CONTINUE
      THETAJK=OBSR(JK)/OBSRST/SUMI
      IF((THETAJK-THETA(JK))**2.GE.TOLERSQ) ISTOP=0
      THETA(JK)=THETAJK
  220 CONTINUE
  230 CONTINUE
```

```
C
      DO 260 I=1,IR
      DO 250 K=1,KT
      KI=K+KT*(I-1)
      SUMJ=0.0
      DO 240 J=1,JS
      IJ=I+IR*(J-1)
      JK=J+JS*(K-1)
      SUMJ=SUMJ+PSI(IJ)*THETA(JK)
  240 CONTINUE
      PHIKI=OBSS(KI)/OBSRST/SUMJ
      IF((PHIKI-PHI(KI))**2.GE.TOLERSQ) ISTOP=0
      PHI(KI)=PHIKI
  250 CONTINUE
  260 CONTINUE
C
      DO 290 J=1,JS
      DO 280 I=1,IR
      IJ=I+IR*(J-1)
      SUMK=0.0
      DO 270 K=1,KT
      JK=J+JS*(K-1)
      KI=K+KT*(I-1)
      SUMK=SUMK+THETA(JK)*PHI(KI)
  270 CONTINUE
      PSIIJ=OBST(IJ)/OBSRST/SUMK
      IF((PSIIJ-PSI(IJ))**2.GE.TOLERSQ) ISTOP=0
      PSI(IJ)=PSIIJ
  280 CONTINUE
  290 CONTINUE
      IF(ITERS.GE.ITERSTOP) GO TO 400
      IF(ISTOP.EQ.0) GO TO 200
      IF(OPTION(4).EQ.0) GO TO 295
      WRITE OUTPUT TAPE OUTPUT,2003
  295 CHISQ=0.0
      DO 300 K=1,KT
      DO 300 J=1,JS
      JK=J+JS*(K-1)
      DO 300 I=1,IR
      IJ=I+IR*(J-1)
      KI=K+KT*(I-1)
      EXPIJK=THETA(JK)*PHI(KI)*PSI(IJ)*OBSRST
      IJK=I+IR*(J-1+JS*(K-1))
      OBSIJK=OBS(IJK)
      CHISQ=CHISQ+(OBSIJK-EXPIJK)*(OBSIJK-EXPIJK)/EXPIJK
      IF(OPTION(4).EQ.0) GO TO 300
      WRITE OUTPUT TAPE OUTPUT,2005,I,J,K,OBSIJK,EXPIJK
  300 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2002
      WRITE OUTPUT TAPE OUTPUT,2004,ITERS,NDEGREES,CHISQ,IR,JS,KT
      GO TO 500
C
  400 WRITE OUTPUT TAPE OUTPUT,2006,ITERSTOP
  500 END
```

```
      SUBROUTINE PLACKETT
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      DIMENSION V(256),VINV(256),Z(16),ZVINV(16)
      TYPE INTEGER OUTPUT
 2000 FORMAT(1H1,10A8)
 2001 FORMAT(20H0PLACKETT PROCEDURE )
 2002 FORMAT(/////19H DEGREES OF FREEDOM ,8X,14H    CHI-SQUARE,17X,
     * 10H SUM(L*L) ,15X,2H R,8X,2H S,8X,2H T)
 2003 FORMAT(/6X,I5,15X,F15.6,11X,F15.6,8X,3(8X,I2))
 2004 FORMAT(3H0K= I2)
 2005 FORMAT(/12H   CONTRASTS )
 2006 FORMAT(/20H   DISPERSION MATRIX )
 2007 FORMAT(/////11H   Z-VECTOR )
 2008 FORMAT(/     11H   P-MATRIX )
      WRITE OUTPUT TAPE OUTPUT,2000,IDENT
      WRITE OUTPUT TAPE OUTPUT,2001
      IR1=IR-1
      JS1=JS-1
      IR1JS1=IR1*JS1
      ZVINVZ=0.0
      DO 30 NAB=1,IR1JS1
      ZVINV(NAB)=0.0
      DO 30 NCD=1,IR1JS1
      NABNCD=NAB+IR1JS1*(NCD-1)
      VINV(NABNCD)=0.0
   30 CONTINUE
      DO 200 K=1,KT
      DO 40 NAB=1,IR1JS1
      Z(NAB)=0.0
      DO 40 NCD=1,IR1JS1
      NABNCD=NAB+IR1JS1*(NCD-1)
      V(NABNCD)=0.0
   40 CONTINUE
      DO 170 J=1,JS
      DO 170 I=1,IR
      IJK=I+IR*(J-1+JS*(K-1))
      OBSIJK=OBS(IJK)
      OBSLOG=LOGF(OBSIJK)
      DO 160 NB=1,JS1
      SIGMA=1.0
      IF(J+NB-JS-1) 60,50,170
   50 SIGMA=-FLOATF(JS-NB)
   60 DO 150 NA=1,IR1
      RHO=1.0
      IF(I+NA-IR-1) 80,70,160
   70 RHO=-FLOATF(IR-NA)
   80 NAB=IR-NA+IR1*(JS1-NB)
      Z(NAB)=Z(NAB)+RHO*SIGMA*OBSLOG
      DO 140 ND=1,JS1
      OMEGA=1.0
```

```
      IF(J+ND-JS-1) 100,90,150
   90 OMEGA=-FLOATF(JS-ND)
  100 DO 130 NC=1,IR1
      TAU=1.0
      IF(1+NC-IR-1) 120,110,140
  110 TAU=-FLOATF(IR-NC)
  120 NCD=IR-NC+IR1*(JS1-ND)
      NABNCD=NAB+IR1JS1*(NCD-1)
      V(NABNCD)=V(NABNCD)+RHO*SIGMA*TAU*OMEGA/OBSIJK
  130 CONTINUE
  140 CONTINUE
  150 CONTINUE
  160 CONTINUE
  170 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2004,K
      WRITE OUTPUT TAPE OUTPUT,2005
      CALL NMPRINT(Z,IR1JS1,1)
      WRITE OUTPUT TAPE OUTPUT,2006
      CALL NMPRINT(V,IR1JS1,IR1JS1)
      CALL MATINV(V,IR1JS1,V,0,D,IR1JS1)
      DO 190 NCD=1,IR1JS1
      ZVINVCD=ZVINV(NCD)
      ZCD=Z(NCD)
      DO 180 NAB=1,IR1JS1
      ZAB=Z(NAB)
      NABNCD=NAB+IR1JS1*(NCD-1)
      VABCD=V(NABNCD)
      ZVINVZ=ZVINVZ+ZAB*VABCD*ZCD
      VINV(NABNCD)=VINV(NABNCD)+VABCD
      ZVINVCD=ZVINVCD+ZAB*VABCD
  180 CONTINUE
      ZVINV(NCD)=ZVINVCD
  190 CONTINUE
  200 CONTINUE
      CALL MATINV(VINV,IR1JS1,V,0,D,IR1JS1)
      CHISQ=ZVINVZ
      DO 210 NCD=1,IR1JS1
      ZVINVCD=ZVINV(NCD)
      DO 210 NAB=1,IR1JS1
      NABNCD=NAB+IR1JS1*(NCD-1)
      CHISQ=CHISQ-ZVINV(NAB)*VINV(NABNCD)*ZVINVCD
  210 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2007
      CALL NMPRINT(ZVINV,IR1JS1,1)
      WRITE OUTPUT TAPE OUTPUT,2008
      CALL NMPRINT(VINV,IR1JS1,IR1JS1)
      WRITE OUTPUT TAPE OUTPUT,2002
      WRITE OUTPUT TAPE OUTPUT,2003,NDEGREES,CHISQ,ZVINVZ,IR,JS,KT
  300 END
```

```
      SUBROUTINE GOODMAN1
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      DIMENSION V(256),VINV(256),Z(16),ZVINV(16)
      TYPE INTEGER OUTPUT
 2000 FORMAT(1H1,10A8)
 2001 FORMAT(19HOGOODMAN PROCEDURE )
 2002 FORMAT(/////19H DEGREES OF FREEDOM ,8X,14H    CHI-SQUARE,17X,
     * 10H SUM(H*H) ,15X,2H R,8X,2H S,8X,2H T)
 2003 FORMAT(/6X,I5,15X,F15.6,11X,F15.6,8X,3(8X,I2))
 2004 FORMAT(3HOK= I2)
 2005 FORMAT(/12H   CONTRASTS )
 2006 FORMAT(/20H   DISPERSION MATRIX )
 2007 FORMAT(/////11H   G-VECTOR )
 2008 FORMAT(/    11H   Q-MATRIX )
      WRITE OUTPUT TAPE OUTPUT,2000,IDENT
      WRITE OUTPUT TAPE OUTPUT,2001
      IR1=IR-1
      JS1=JS-1
      IR1JS1=IR1*JS1
      ZVINVZ=0.0
      DO 30 NAB=1,IR1JS1
      ZVINV(NAB)=0.0
      DO 30 NCD=1,IR1JS1
      NABNCD=NAB+IR1JS1*(NCD-1)
      VINV(NABNCD)=0.0
   30 CONTINUE
      DO 200 K=1,KT
      DO 40 NAB=1,IR1JS1
      Z(NAB)=0.0
      DO 40 NCD=1,IR1JS1
      NABNCD=NAB+IR1JS1*(NCD-1)
      V(NABNCD)=0.0
   40 CONTINUE
      DO 170 J=1,JS
      DO 170 I=1,IR
      IJK=I+IR*(J-1+JS*(K-1))
      OBSIJK=OBS(IJK)
      OBSLOG=LOGF(OBSIJK)
      DO 160 NB=1,JS1
      SIGMA=-1.0
      IF(J.EQ.JS) GO TO 60
      IF(NB.NE.J) GO TO 160
      SIGMA=1.0
   60 DO 150 NA=1,IR1
      RHO=-1.0
      IF(I.EQ.IR) GO TO 80
      IF(NA.NE.I) GO TO 150
      RHO=1.0
   80 NAB=NA+IR1*(NB-1)
      Z(NAB)=Z(NAB)+RHO*SIGMA*OBSLOG
```

```
      DO 140 ND=1,JS1
      OMEGA=-1.0
      IF(J.EQ.JS) GO TO 100
      IF(ND.NE.J) GO TO 140
      OMEGA=1.0
  100 DO 130 NC=1,IR1
      TAU=-1.0
      IF(I.EQ.IR) GO TO 120
      IF(NC.NE.I) GO TO 130
      TAU=1.0
  120 NCD=NC+IR1*(ND-1)
      NABNCD=NAB+IR1JS1*(NCD-1)
      V(NABNCD)=V(NABNCD)+RHO*SIGMA*TAU*OMEGA/OBSIJK
  130 CONTINUE
  140 CONTINUE
  150 CONTINUE
  160 CONTINUE
  170 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2004,K
      WRITE OUTPUT TAPE OUTPUT,2005
      CALL NMPRINT(Z,IR1JS1,1)
      WRITE OUTPUT TAPE OUTPUT,2006
      CALL NMPRINT(V,IR1JS1,IR1JS1)
      CALL MATINV(V,IR1JS1,V,0,D,IR1JS1)
      DO 190 NCD=1,IR1JS1
      ZVINVCD=ZVINV(NCD)
      ZCD=Z(NCD)
      DO 180 NAB=1,IR1JS1
      ZAB=Z(NAB)
      NABNCD=NAB+IR1JS1*(NCD-1)
      VABCD=V(NABNCD)
      ZVINVZ=ZVINVZ+ZAB*VABCD*ZCD
      VINV(NABNCD)=VINV(NABNCD)+VABCD
      ZVINVCD=ZVINVCD+ZAB*VABCD
  180 CONTINUE
      ZVINV(NCD)=ZVINVCD
  190 CONTINUE
  200 CONTINUE
      CALL MATINV(VINV,IR1JS1,V,0,D,IR1JS1)
      CHISQ=ZVINVZ
      DO 210 NCD=1,IR1JS1
      ZVINVCD=ZVINV(NCD)
      DO 210 NAB=1,IR1JS1
      NABNCD=NAB+IR1JS1*(NCD-1)
      CHISQ=CHISQ-ZVINV(NAB)*VINV(NABNCD)*ZVINVCD
  210 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2007
      CALL NMPRINT(ZVINV,IR1JS1,1)
      WRITE OUTPUT TAPE OUTPUT,2008
      CALL NMPRINT(VINV,IR1JS1,IR1JS1)
      WRITE OUTPUT TAPE OUTPUT,2002
      WRITE OUTPUT TAPE OUTPUT,2003,NDEGREES,CHISQ,ZVINVZ,IR,JS,KT
  300 END
```

```
      SUBROUTINE GOODMAN2
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      DIMENSION P(256),Q(256),B(80),C(16),G(16),PG(16)
      TYPE INTEGER OUTPUT
 2000 FORMAT(1H1,10A8)
 2001 FORMAT(28HOMODIFIED GOODMAN PROCEDURE ,////1H )
 2002 FORMAT(/12H   CONTRASTS )
 2003 FORMAT(/////19H DEGREES OF FREEDOM ,8X,14H    CHI-SQUARE,15X,2H F
     *   8X,2H S,8X,2H T )
 2004 FORMAT(/6X,I5,15X,F15.6,7X,3(8X,I2))
      WRITE OUTPUT TAPE OUTPUT,2000,IDENT
      WRITE OUTPUT TAPE OUTPUT,2001
      IR1=IR-1
      JS1=JS-1
      IR1IR1=IR1*IR1
      IR1JS1=IR1*JS1
      CHISQ=0.0
      DO 15 NAB=1,IR1JS1
      PG(NAB)=0.0
      DO 10 NCD=1,IR1JS1
      NABNCD=NAB+IR1JS1*(NCD-1)
      Q(NABNCD)=0.0
   10 CONTINUE
   15 CONTINUE
      DO 200 K=1,KT
      DO 25 NAB=1,IR1JS1
      G(NAB)=0.0
      DO 20 NCD=1,IR1JS1
      NABNCD=NAB+IR1JS1*(NCD-1)
      P(NABNCD)=0.0
   20 CONTINUE
   25 CONTINUE
      DO 35 II=1,IR1
      DO 30 III=1,IR1
      IIIII=III+IR1*(II-1)
      C(IIIII)=0.0
   30 CONTINUE
   35 CONTINUE
      DO 100 J=1,JS
      JK=IR*(J-1+JS*(K-1))
      OBSJK=0.0
      DO 70 I=1,IR
      IJK=I+JK
      OBSIJK=OBS(IJK)
      OBSLOG=LOGF(OBSIJK)
      DO 60 NB=1,JS1
      SIGMA=-1.0
      IF(J.EQ.JS) GO TO 40
      IF(NB.NE.J) GO TO 60
      SIGMA=1.0
```

```
 40 DO 50 NA=1,IR1
    RHO=-1.0
    IF(I.EQ.IR) GO TO 45
    IF(NA.NE.I) GO TO 50
    RHO=1.0
 45 NAB=NA+IR1*(NB-1)
    G(NAB)=G(NAB)+RHO*SIGMA*OBSLOG
 50 CONTINUE
 60 CONTINUE
    OBSJK=OBSJK+OBSIJK
 70 CONTINUE
    DO 90 II=1,IR1
    IIJK=II+JK
    DO 80 III=1,IR1
    IIIJK=III+JK
    IIIII=III+IR1*(II-1)
    IIIIIJ=IIIII+IR1IR1*(J-1)
    BIIIIIJ=-OBS(IIJK)*OBS(IIIJK)/OBSJK
    IF(II.EQ.III) BIIIIIJ=BIIIIIJ+OBS(IIJK)
    C(IIIII)=C(IIIII)+BIIIIIJ
    B(IIIIIJ)=BIIIIIJ
 80 CONTINUE
 90 CONTINUE
100 CONTINUE
    WRITE OUTPUT TAPE OUTPUT,2002
    CALL NMPRINT(G,IR1JS1,1)
    CALL MATINV(C,IR1,C,0,D,IR1)
    DO 170 NB=1,JS1
    DO 160 ND=1,JS1
    DO 150 NA=1,IR1
    NAB=NA+IR1*(NB-1)
    DO 140 NC=1,IR1
    NCD=NC+IR1*(ND-1)
    NABNCD=NAB+IR1JS1*(NCD-1)
    DO 120 II=1,IR1
    IINCND=II+IR1*(NC-1+IR1*(ND-1))
    DO 110 III=1,IR1
    NAIIINB=NA+IR1*(III-1+IR1*(NB-1))
    IIIII=III+IR1*(II-1)
    P(NABNCD)=P(NABNCD)-B(NAIIINB)*C(IIIII)*B(IINCND)
110 CONTINUE
120 CONTINUE
    IF(NB.NE.ND) GO TO 130
    NANCND=NA+IR1*(NC-1+IR1*(ND-1))
    P(NABNCD)=P(NABNCD)+B(NANCND)
130 Q(NABNCD)=Q(NABNCD)+P(NABNCD)
    PG(NAB)=PG(NAB)+P(NABNCD)*G(NCD)
140 CONTINUE
150 CONTINUE
160 CONTINUE
```

```
170 CONTINUE
    DO 190 NAB=1,IR1JS1
    DO 180 NCD=1,IR1JS1
    NABNCD=NAB+IR1JS1*(NCD-1)
    CHISQ=CHISQ+G(NAB)*P(NABNCD)*G(NCD)
180 CONTINUE
190 CONTINUE
200 CONTINUE
    CALL MATINV(Q,IR1JS1,Q,0,D,IR1JS1)
    DO 220,NAB=1,IR1JS1
    DO 210 NCD=1,IR1JS1
    NABNCD=NAB+IR1JS1*(NCD-1)
    CHISQ=CHISQ-PG(NAB)*Q(NABNCD)*PG(NCD)
210 CONTINUE
220 CONTINUE
    WRITE OUTPUT TAPE OUTPUT,2003
    WRITE OUTPUT TAPE OUTPUT,2004,NDEGREES,CHISQ,IR,JS,KT
300 END
```

```
      SUBROUTINE TWOBYTWO
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      DIMENSION D(16),V(16),G(16),U(16),B(16),A(16),DSTAR(16),VSTAR(16),
     *   GSTAR(16),USTAR(16),BSTAR(16),ASTAR(16)
      TYPE INTEGER OUTPUT
 2000 FORMAT(1H1,10A8)
 2001 FORMAT(26H0GOODMANS 2X2XT PROCEDURE ,////1H )
 2002 FORMAT(3H0 K,8X,3H D ,10X,3H V ,10X,3HX*X,10X,3H B ,10X,3H A ,10X,
     * 3HZ*Z,10X,3H G ,10X,3H U ,10X,3HY*Y ,/1H )
 2003 FORMAT(I3,9F13.5)
 2004 FORMAT( I3,3(2F13.5,13X))
 2005 FORMAT(/////19H DEGREES OF FREEDOM ,11X,3HX*X,15X,3HZ*Z,15X,3HY*Y,
     * 15X,2H R,8X,2H S,8X,2H T,/1H )
 2006 FORMAT(6X,I5,6X,3F18.5,5X,3(8X,I2))
 2007 FORMAT(3X,3(26X,F13.5))
      WRITE OUTPUT TAPE OUTPUT,2000,IDENT
      WRITE OUTPUT TAPE OUTPUT,2001
      IF(IR.NE.2) GO TO 100
      IF(JS.NE.2) GO TO 100
      WRITE OUTPUT TAPE OUTPUT,2002
      SUM1=SUM2=SUM3=SUM4=SUM5=SUM6=SUM7=SUM8=SUM9=0.0
      DO 20 K=1,KT
      KPART=IR*JS*(K-1)
      OBS11K=OBS(1+KPART)
      OBS21K=OBS(2+KPART)
      OBS12K=OBS(3+KPART)
      OBS22K=OBS(4+KPART)
      DK=OBS11K*OBS22K/OBS12K/OBS21K
      UK=1.0/OBS11K+1.0/OBS22K+1.0/OBS12K+1.0/OBS21K
      VK=DK*DK*UK
      WK=1.0/VK
      GK=LOGF(DK)
      PK=1.0/UK
      BK=1.0/DK
      AK=BK*BK*UK
      HK=1.0/AK
      SUM1=SUM1+DK*DK*WK
      SUM2=SUM2+DK*WK
      SUM3=SUM3+WK
      SUM4=SUM4+GK*GK*PK
      SUM5=SUM5+GK*PK
      SUM6=SUM6+PK
      SUM7=SUM7+BK*BK*HK
      SUM8=SUM8+BK*HK
      SUM9=SUM9+HK
      DSTAR(K)=SUM2/SUM3
      VSTAR(K)=1.0/SUM3
```

#636

```
      GSTAR(K)=SUM5/SUM6
      USTAR(K)=1.0/SUM6
      BSTAR(K)=SUM8/SUM9
      ASTAR(K)=1.0/SUM9
      D(K)=DK
      V(K)=VK
      G(K)=GK
      U(K)=UK
      B(K)=BK
      A(K)=AK
   20 CONTINUE
      SUMXSQ=SUMYSQ=SUMZSQ=0.0
      KT1=KT-1
      DO 30 K=1,KT1
      XSQK=(D(K+1)-DSTAR(K))*(D(K+1)-DSTAR(K))/(V(K+1)+VSTAR(K))
      YSQK=(G(K+1)-GSTAR(K))*(G(K+1)-GSTAR(K))/(U(K+1)+USTAR(K))
      ZSQK=(B(K+1)-BSTAR(K))*(B(K+1)-BSTAR(K))/(A(K+1)+ASTAR(K))
      WRITE OUTPUT TAPE OUTPUT,2003,K,D(K),V(K),XSQK,B(K),A(K),ZSQK,
     * G(K),U(K),YSQK
      SUMXSQ=SUMXSQ+XSQK
      SUMYSQ=SUMYSQ+YSQK
      SUMZSQ=SUMZSQ+ZSQK
   30 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2004,KT,D(KT),V(KT),B(KT),A(KT),G(KT),
     * U(KT)
      WRITE OUTPUT TAPE OUTPUT,2007,SUMXSQ,SUMZSQ,SUMYSQ
      XXSQ=SUM1-SUM2*SUM2/SUM3
      YYSQ=SUM4-SUM5*SUM5/SUM6
      ZZSQ=SUM7-SUM8*SUM8/SUM9
      WRITE OUTPUT TAPE OUTPUT,2005
      WRITE OUTPUT TAPE OUTPUT,2006,NDEGREES,XXSQ,ZZSQ,YYSQ,IR,JS,KT
  100 END
```


#636

```
      SUBROUTINE KKK
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      COMMON/SUMS/OBSR(80),OBSS(80),OBST(25),OBSRS(16),OBSTR(5),
     *   OBSST(5),OBSRST
      TYPE INTEGER OUTPUT
 2000 FORMAT(1H1,10A8)
 2001 FORMAT(39HOKULLBACK, KUPPERMAN, AND KU PROCEDURE )
 2002 FORMAT(/////19H DEGREES OF FREEDOM ,11X,3H2*I,17X,2H R,8X,2H S,8X,
     * 2H T)
 2003 FORMAT(/6X,I5,6X,F18.5,7X,3(8X,I2))
      WRITE OUTPUT TAPE OUTPUT,2000,IDENT
      WRITE OUTPUT TAPE OUTPUT,2001
      CALL GETSUMS
      TWOI=0.0
      DO 140 K=1,KT
      OBSRSK=OBSRS(K)
      DO 130 J=1,JS
      OBSTRJ=OBSTR(J)
      JK=J+JS*(K-1)
      OBSRJK=OBSR(JK)
      DO 120 I=1,IR
      OBSSTI=OBSST(I)
      IJ=I+IR*(J-1)
      OBSTIJ=OBST(IJ)
      KI=K+KT*(I-1)
      OBSSKI=OBSS(KI)
      IJK=I+IR*(J-1+JS*(K-1))
      OBSIJK=OBS(IJK)
      TWOI=TWOI+OBSIJK*LOGF(OBSIJK*OBSSTI*OBSTRJ*OBSRSK/OBSRST/OBSTIJ/
     *   OBSSKI/OBSRJK)
  120 CONTINUE
  130 CONTINUE
  140 CONTINUE
      TWOI=2.0*TWOI
      WRITE OUTPUT TAPE OUTPUT,2002
      WRITE OUTPUT TAPE OUTPUT,2003,NDEGREES,TWOI,IR,JS,KT
  400 END
```

```
      SUBROUTINE INTRVALS
      COMMON/BCD/IBVECTOR(20),JCVECTOR(20),KDVECTOR(20)
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      COMMON/FLAGS/OPTION(10),LOGFLAG
      DIMENSION Q(6),CHIT(2),CHI1(2),PHIW(2),P(2)
      TYPE INTEGER OUTPUT
      TYPE INTEGER OPTION
      DATA(P=0.95,0.99)
 2000 FORMAT(1H1,10A8)
 2001 FORMAT(61H0INTERACTION ESTIMATES AND CONFIDENCE INTERVALS IN LOG U
     *NITS )
 2002 FORMAT(63H0INTERACTION ESTIMATES AND CONFIDENCE INTERVALS IN BASIC
     * UNITS )
 2003 FORMAT(//// 44H           B                   C                   D    )
 2004 FORMAT(/9X,I2,14X,I2,14X,I2)
 2005 FORMAT(//// 34H           P                   CHI(T=I5,43H)
     *     CHI(1)                 PHI(W=I5,1H),/1H )
 2006 FORMAT(9X,F4.2,3(9X,F15.4))
 2007 FORMAT(/////106H0 I  B  J  C  K  D       P           ESTIMATE        S
     *TANDARD                                   ...LIMITS... /,50X,77H
     *ERROR                  CHI(T)                CHI(1)
     *   PHI(W) /,1H )
 2008 FORMAT(6I3,F9.2,2E15.3,4X,3(4X,2E10.2))
 2009 FORMAT(/////106H0 I  B  J  C  K  D       P           ESTIMATE
     *                                          ...LIMITS... /,72X,55HC
     *HI(T)                  CHI(1)                PHI(W) /,1H )
 2010 FORMAT(6I3,F9.2,E15.3,15X,4X,3(4X,2E10.2))
      WRITE OUTPUT TAPE OUTPUT,2000,IDENT
      IF(LOGFLAG.EQ.0) GO TO 5
      WRITE OUTPUT TAPE OUTPUT,2001
      GO TO 10
    5 WRITE OUTPUT TAPE OUTPUT,2002
   10 WRITE OUTPUT TAPE OUTPUT,2003
      NW=0
      NBCD=0
      DO 15 I=1,20
      IB=IBVECTOR(I)
      IF(IB.EQ.0) GO TO 15
      JC=JCVECTOR(I)
      KD=KDVECTOR(I)
      NBCD=NBCD+1
      NW=NW+(IB-1)*(JC-1)*(KD-1)
      WRITE OUTPUT TAPE OUTPUT,2004,IB,JC,KD
   15 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2005,NDEGREES,NW
      W=FLOATF(NW)
      T=FLOATF(NDEGREES)
      DO 20 L=1,2
      PL=P(L)
      CHITL=CHI(T,PL)
```

```
      CHI1L=CHI(1.0,PL)
      PHIWL=PHI(W,PL)
      WRITE OUTPUT TAPE OUTPUT,2006,PL,CHITL,CHI1L,PHIWL
      CHIT(L)=CHITL
      CHI1(L)=CHI1L
      PHIW(L)=PHIWL
   20 CONTINUE
      IF(LOGFLAG.EQ.0) GO TO 25
      WRITE OUTPUT TAPE OUTPUT,2007
      GO TO 30
   25 WRITE OUTPUT TAPE OUTPUT,2009
   30 DO 100 N=1,NBCD
      KD=KDVECTOR(N)
      KD1=KD-1
      JC=JCVECTOR(N)
      JC1=JC-1
      IB=IBVECTOR(N)
      IB1=IB-1
      DO 70 K=1,KD1
      DO 60 J=1,JC1
      DO 50 I=1,IB1
      IJK=I+IR*(J-1+JS*(K-1))
      OBSIJK=OBS(IJK)
      IBJCK=IB+IR*(JC-1+JS*(K-1))
      OBSBCK=OBS(IBJCK)
      IBJK=IB+IR*(J-1+JS*(K-1))
      OBSBJK=OBS(IBJK)
      IJCK=I+IR*(JC-1+JS*(K-1))
      OBSICK=OBS(IJCK)
      IJKD=I+IR*(J-1+JS*(KD-1))
      OBSIJD=OBS(IJKD)
      IBJCKD=IB+IR*(JC-1+JS*(KD-1))
      OBSBCD=OBS(IBJCKD)
      IBJKD=IB+IR*(J-1+JS*(KD-1))
      OBSBJD=OBS(IBJKD)
      IJCKD=I+IR*(JC-1+JS*(KD-1))
      OBSICD=OBS(IJCKD)
      AIBJCKD=LOGF(OBSIJK*OBSBCK*OBSBJD*OBSICD/OBSBJK/OBSICK/OBSIJD/
     *                 OBSBCD)
      UIBJCK=1.0/OBSIJK+1.0/OBSBCK+1.0/OBSICK+1.0/OBSBJK
      UIBJCD=1.0/OBSIJD+1.0/OBSBCD+1.0/OBSICD+1.0/OBSBJD
      SIBJCKD=SQRTF(UIBJCK+UIBJCD)
      DO 45 L=1,2
      PL=P(L)
      Q(1)=AIBJCKD-SIBJCKD*CHIT(L)
      Q(2)=2.0*AIBJCKD-Q(1)
      Q(3)=AIBJCKD-SIBJCKD*CHI1(L)
      Q(4)=2.0*AIBJCKD-Q(3)
      Q(5)=AIBJCKD-SIBJCKD*PHIW(L)
```

```
      Q(6)=2.0*AIBJCKD-Q(5)
      IF(LOGFLAG.EQ.0) GO TO 35
      WRITE OUTPUT TAPE OUTPUT,2008,I,IB,J,JC,K,KD,PL,AIBJCKD,SIBJCKD,Q
      GO TO 45
   35 AIBJCKD=EXPF(AIBJCKD)
      DO 40 M=1,6
      Q(M)=EXPF(Q(M))
   40 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2010,I,IB,J,JC,K,KD,PL,AIBJCKD,Q
   45 CONTINUE
   50 CONTINUE
   60 CONTINUF
   70 CONTINUE
  100 CONTINUE
      END
```

```
      SUBROUTINE READIN
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREFS,INPUT,OUTPUT
      COMMON/BCD/IBVECTOR(20),JCVECTOR(20),KDVECTOR(20)
      COMMON/FLAGS/OPTION(10),LOGFLAG
      TYPE INTEGER OPTION
      TYPE INTEGER OUTPUT
      DIMENSION INMAT(10)
 1000 FORMAT(10A8)
 1001 FORMAT(20I4)
 1002 FORMAT(10I1)
      READ INPUT TAPE INPUT,1000,IDENT
      IF(IDENT(1).EQ.8HTHATSALL) GO TO 400
      READ INPUT TAPE INPUT,1002,OPTION
      READ INPUT TAPE INPUT,1001,IR,JS,KT
      READ INPUT TAPE INPUT,1000,INMAT
      NRST=IR*JS*KT
      READ INPUT TAPE INPUT,INMAT,(OBS(N),N=1,NRST)
      READ INPUT TAPE INPUT,1001,IBVECTOR
      READ INPUT TAPE INPUT,1001,JCVECTOR
      READ INPUT TAPE INPUT,1001,KDVECTOR
      NDEGREFS=(IR-1)*(JS-1)*(KT-1)
      RETURN
  400 STOP
      END
```

```
      SUBROUTINE GETSUMS
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      COMMON/SUMS/OBSR(80),OBSS(80),OBST(25),OBSRS(16),OBSTR(5),
     *   OBSST(5),OBSRST
      OBSRST=0.0
      DO 30 K=1,KT
      OBSRSK=0.0
      DO 20 J=1,JS
      OBSRJK=0.0
      DO 10 I=1,IR
      IJK=I+IR*(J-1+JS*(K-1))
      OBSIJK=OBS(IJK)
      OBSRJK=OBSRJK+OBSIJK
      OBSRSK=OBSRSK+OBSIJK
      OBSRST=OBSRST+OBSIJK
   10 CONTINUE
      JK=J+JS*(K-1)
      OBSR(JK)=OBSRJK
   20 CONTINUE
      OBSRS(K)=OBSRSK
   30 CONTINUE
C
      DO 60 J=1,JS
      OBSTRJ=0.0
      DO 50 I=1,IR
      OBSTIJ=0.0
      DO 40 K=1,KT
      IJK=I+IR*(J-1+JS*(K-1))
      OBSIJK=OBS(IJK)
      OBSTIJ=OBSTIJ+OBSIJK
      OBSTRJ=OBSTRJ+OBSIJK
   40 CONTINUE
      IJ=I+IR*(J-1)
      OBST(IJ)=OBSTIJ
   50 CONTINUE
      OBSTR(J)=OBSTRJ
   60 CONTINUE
C
      DO 90 I=1,IR
      OBSSTI=0.0
      DO 80 K=1,KT
      OBSSKI=0.0
      DO 70 J=1,JS
      IJK=I+IR*(J-1+JS*(K-1))
      OBSIJK=OBS(IJK)
      OBSSKI=OBSSKI+OBSIJK
      OBSSTI=OBSSTI+OBSIJK
   70 CONTINUE
      KI=K+KT*(I-1)
      OBSS(KI)=OBSSKI
   80 CONTINUE
      OBSST(I)=OBSSTI
   90 CONTINUE
      END
```

```
      SUBROUTINE NMPRINT(X,N,M)
      COMMON/ALL/OBS(400),IDENT(10),IR,JS,KT,NDEGREES,INPUT,OUTPUT
      DIMENSION X(256),XNORM(256)
      TYPE INTEGER OUTPUT
 2000 FORMAT(/9X,32HALL NUMBERS HAVE BEEN DIVIDED BY E11.1/,1H )
 2001 FORMAT(8X,16F7.3)
      NM=N*M
      XMAX=X(1)
      DO 10 K=2,NM
      IF(XMAX*XMAX.GT.X(K)*X(K)) GO TO 10
      XMAX=X(K)
   10 CONTINUE
      XMAX=ABSF(XMAX)
      XMAXLOG=LOGF(XMAX)/2.3025850930
      MAXLOG=XMAXLOG
      IF(XMAX.LT.1.0) MAXLOG=MAXLOG-1
      XMAX=10.0**MAXLOG
      DO 20 K=1,NM
      XNORM(K)=X(K)/XMAX
   20 CONTINUE
      WRITE OUTPUT TAPE OUTPUT,2000,XMAX
      DO 30 I=1,M
      J2=N*I
      J1=J2-N+1
      WRITE OUTPUT TAPE OUTPUT,2001,(XNORM(J),J=J1,J2)
   30 CONTINUE
      END
```

```
      FUNCTION PHI(W,PL)
      AREA=1.0+(PL-1.0)/2.0/W
      TEST=0.
      EP = .0000001
      IF(AREA-.5)3,4,5
    4 XORD=0.
      GOTO16
    3 AREA1=1.-AREA
      C = -1.
      GOTO7
    5 AREA1=AREA
      C = 1.
    7 X = (AREA1 - .5) * 2.5
    8 DIV = 1. + .3275911 * X
      E = 1.0/ DIV
      S =  ((((.940646070*E)-1.287822453)*E+1.259695130)*E-.252128668)*E
     X+.225836846
      SD = ((((( 4.70323035      * E ) - 5.151289812      )*E +
     X3.77908539  ) * E - .504257336      ) * E + .225836846 ) * E
      AA = EXPF (    X **2 ) * .88622692
      FX = (AREA1 - 1. ) * AA * DIV * 2. + S
      FXP = - SD * .3275911      - S * 2. * X
      XNEW = X - FX / FXP
      IF(ABSF (XNEW-X)    -EP ) 6, 6, 14
   14 TEST=TEST+1.
      IF(TEST-40.)15,6,6
   15 X = XNEW
      GO TO 8
    6 XORD = C * XNEW * 1.414213562
   16 PHI=XORD
      END
```

[redacted]

```
      FUNCTION CHI(T,PL)
      DIMENSION X(13,2)
      DATA(X=3.841,5.991,7.815,9.488,11.070,1.000,2.32638131,1.13735,
     * -0.554967955,-0.123011797,0.077939867,-0.100617878,0.122451698,
     * 6.635,9.210,11.345,13.277,15.086,1.000,3.28946075,2.94018400,
     * -0.290518594,-0.541655265,0.411341798,-0.342315613,0.203111198)
      L=1
      IF(PL.GT.0.975) L=2
      IF(T.GT.5.5) GO TO 10
      IT=T
      CHI=X(IT,L)
      GO TO 30
   10 TRT=SQRTF(T)
      CHI=X(13,L)
      DO 20 K=1,7
      CHI=CHI/TRT+X(13-K,L)
   20 CONTINUE
      CHI=T*CHI
   30 CHI=SQRTF(CHI)
      END
```

#636

TABLE 1 - DATA FROM KASTENBAUM AND LAMPHIEAR (1959) - 2 X 3 X 5

KASTENBAUM-LAMPHIEAR PROCEDURE

CELL			OBSERVED FREQUENCY	EXPECTED FREQUENCY
1	1	1	58.000000	54.995024
2	1	1	75.000000	78.004976
1	2	1	11.000000	12.381343
2	2	1	19.000000	17.618657
1	3	1	5.000000	6.623632
2	3	1	7.000000	5.376368
1	1	2	49.000000	48.379034
2	1	2	58.000000	58.620966
1	2	2	14.000000	13.991513
2	2	2	17.000000	17.008487
1	3	2	10.000000	10.629453
2	3	2	8.000000	7.370547
1	1	3	33.000000	34.127160
2	1	3	45.000000	43.872840
1	2	3	18.000000	17.469260
2	2	3	22.000000	22.530740
1	3	3	15.000000	14.403580
2	3	3	10.000000	10.596420
1	1	4	15.000000	17.132082
2	1	4	39.000000	36.867918
1	2	4	13.000000	11.079608
2	2	4	22.000000	23.920392
1	3	4	15.000000	14.788311
2	3	4	18.000000	18.211689
1	1	5	4.000000	4.366700
2	1	5	5.000000	4.633300
1	2	5	12.000000	13.078277
2	2	5	15.000000	13.921723
1	3	5	17.000000	15.555023
2	3	5	8.000000	9.444977

ITERATION	DEGREES OF FREEDOM	CHI-SQUARE
26	8	3.158045

R	S	T
2	3	5

#636

TABLE 2 - DATA FROM KASTENBAUM AND LAMPHIEAR (1959) - 2 X 3 X 5

PLACKETT PROCEDURE

K= 1

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

2.895 -1.306

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0+000

0.174	-0.113
-0.113	1.546

K= 2

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

0.255 -8.091

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0+000

0.158	-0.093
-0.093	1.068

K= 3

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0+000

-0.139 -1.322

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

1.535	-0.485
-0.485	8.202

TABLE 2 (Continued)

K= 4

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0+000

-0.429 -1.117

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

2.147	-0.301
-0.301	7.036

K= 5

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0+000

0.000 -1.954

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0+000

0.600	0.300
0.300	1.335

Z-VECTOR

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0+000

-1.248 -5.760

P-MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

0.336	-0.088
-0.088	1.951

DEGREES OF FREEDOM	CHI-SQUARE	SUM(L*L)
8	3.127641	9.533446

R	S	T
2	3	5

TABLE 3 - DATA FROM KASTENBAUM AND LAMPHIEAR (1959) - 2 X 3 X 5

GOODMAN PROCEDURE

K= 1

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

0.794 -2.101

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

3.734 3.429
3.429 4.964

K= 2

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

-3.918 -4.173

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

2.626 2.250
2.250 3.553

K= 3

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

-7.156 -6.061

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

2.192 1.667
1.667 2.677

#636

TABLE 3 (Continued)

K= 4

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

-7.732 -3.438

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

2.145	1.222
1.222	2.446

K= 5

CONTRASTS

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

-9.769 -9.769

DISPERSION MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-001

6.338	1.838
1.838	3.338

G-VECTOR

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0+000

-7.008 -4.511

Q-MATRIX

ALL NUMBERS HAVE BEEN DIVIDED BY 1.0-002

5.426	3.886
3.886	6.309

DEGREES OF FREEDOM	CHI-SQUARE	SUM(H*H)
8	3.127641	9.533446

R	S	T
2	3	5

#636

[redacted]

TABLE 4 - DATA FROM NORTON (1945) - 2 X 2 X 12

GOODMANS 2X2XT PROCEDURE

K	D	V	X*X	B	A
1	0.05543	0.00084	0.29711	18.03922	88.58704
2	0.03616	0.00041	0.02239	27.65714	242.38946
3	0.04712	0.00066	0.21299	21.22105	133.87251
4	0.06208	0.00136	0.96746	16.10714	91.40393
5	0.02715	0.00020	0.00713	36.83333	375.06906
6	0.03947	0.00045	0.03231	25.33333	184.97354
7	0.03393	0.00040	0.95887	29.47368	300.12538
8	0.07353	0.00131	0.92882	13.60000	44.85733
9	0.07045	0.00101	0.14199	14.19444	40.97770
10	0.03316	0.00035	1.69991	30.16000	285.59509
11	0.01831	0.00022	2.18260	54.62500	1940.89453
12	0.01655	0.00013		60.41667	1695.44271
			7.45158		

Z*Z	G	U	Y*Y
0.27949	-2.89255	0.27223	0.30999
0.00186	-3.31988	0.31688	0.00277
0.16383	-3.05499	0.29727	0.19875
0.76060	-2.77926	0.35231	0.99599
0.10675	-3.60640	0.276[illegible]	0.02324
0.21198	-3.23212	0.28822	0.12904
1.00329	-3.38350	0.34549	1.16016
0.42970	-2.61007	0.24252	0.82690
0.51606	-2.65285	0.20338	0.41032
0.67826	-3.40652	0.31397	1.29140
1.03293	-4.00049	0.65046	2.02544
	-4.10127	0.46448	
5.18475			7.37399

DEGREES OF FREEDOM	X*X	Z*Z	Y*Y
11	7.45158	5.18475	7.37399

Q	S	T
2	2	12

#636

TABLE 5 - DATA FROM KASTENBAUM AND LAMPHIEAR (1959) - 2 X 3 X 5

INTERACTION ESTIMATES AND CONFIDENCE INTERVALS IN LOG UNITS

B	C	D
2	3	5

P	CHI(T= 8)	CHI(1)
0.95	3.9381	1.9598
0.99	4.4819	2.5758

I	B	J	C	K	D	P	ESTIMATE	STANDARD ERROR
1	2	1	3	1	5	0.95	1.056+000	1.004+000
1	2	1	3	1	5	0.99	1.056+000	1.004+000
1	2	2	3	1	5	0.95	7.668-001	9.057-001
1	2	2	3	1	5	0.99	7.668-001	9.057-001
1	2	1	3	2	5	0.95	5.851-001	9.468-001
1	2	1	3	2	5	0.99	5.851-001	9.468-001
1	2	2	3	2	5	0.95	5.596-001	8.301-001
1	2	2	3	2	5	0.99	5.596-001	8.301-001
1	2	1	3	3	5	0.95	2.613-001	9.236-001
1	2	1	3	3	5	0.99	2.613-001	9.236-001
1	2	2	3	3	5	0.95	3.708-001	7.756-001
1	2	2	3	3	5	0.99	3.708-001	7.756-001
1	2	1	3	4	5	0.95	2.037-001	9.211-001
1	2	1	3	4	5	0.99	2.037-001	9.211-001
1	2	2	3	4	5	0.95	6.331-001	7.605-001
1	2	2	3	4	5	0.99	6.331-001	7.605-001

#636

TABLE 5 (Continued)

PHI(W= 8)

2.7344
3.2272

...LIMITS...

CHI(T)		CHI(1)		PHI(W)	
-2.90+000	5.01+000	-9.11-001	3.02+000	-1.69+000	3.80+00
-3.44+000	5.55+000	-1.53+000	3.64+000	-2.18+000	4.30+00
-2.80+000	4.33+000	-1.01+000	2.54+000	-1.71+000	3.24+00
-3.29+000	4.83+000	-1.57+000	3.10+000	-2.16+000	3.69+00
-3.14+000	4.31+000	-1.27+000	2.44+000	-2.00+000	3.17+00
-3.66+000	4.83+000	-1.85+000	3.02+000	-2.47+000	3.64+00
-2.71+000	3.83+000	-1.07+000	2.19+000	-1.71+000	2.83+00
-3.16+000	4.28+000	-1.58+000	2.70+000	-2.12+000	3.24+00
-3.38+000	3.90+000	-1.55+000	2.07+000	-2.26+000	2.79+00
-3.88+000	4.40+000	-2.12+000	2.64+000	-2.72+000	3.24+00
-2.68+000	3.42+000	-1.15+000	1.89+000	-1.75+000	2.49+00
-3.11+000	3.85+000	-1.63+000	2.37+000	-2.13+000	2.87+00
-3.42+000	3.83+000	-1.60+000	2.01+000	-2.31+000	2.72+00
-3.92+000	4.33+000	-2.17+000	2.58+000	-2.77+000	3.18+00
-2.36+000	3.63+000	-8.57-001	2.12+000	-1.45+000	2.71+00
-2.78+000	4.04+000	-1.33+000	2.59+000	-1.82+000	3.09+00